基于椭圆曲线的高效分级群签名

王国才, 刘美兰
(中南大学 信息科学与工程学院, 长沙 410083)

摘 要: 为了满足现代电子商务和电子政务的高性能需求,提出一种高效的分级群签名方案。方案通过对椭圆曲线签名方案进行改进,避免了耗时的模逆、模乘运算,并减少了一次点乘运算,提高了签名和验证算法的效率。在此基础上构造了一个高效的群签名方案,引入消息等级表,提出了一个基于椭圆曲线的高效分级群签名方案。经分析表明,该方案大大缩短了分级群签名和验证的时间,与现有方案相比,具有更高的效率和安全性,同时具有椭圆曲线密码体制的优点,适用于智能系统中,实用性强。

关键词: 分级群签名; 群签名; 椭圆曲线; 权限; 知识签名
中图分类号: TP393.08 **文献标志码**: A **文章编号**: 1001-3695(2014)02-0586-04
doi:10.3969/j.issn.1001-3695.2014.02.064

Efficient hierarchical group signature based on elliptic curve

WANG Guo-cai, LIU Mei-lan
(*College of Information Science & Engineering*, *Central South University*, *Changsha* 410083, *China*)

Abstract: This paper presented an efficient hierarchical group signature scheme which could meet the high performance needs of modern e-commerce and e-government. First, the scheme improved the elliptic curve digital signature scheme, which avoided time-consuming modular inverse, the modular multiplication and reduced once point multiplication. Thus it improved the efficiency of the signature and verification algorithms. Then the scheme constructed an efficient group signature scheme based on the improved ECDSA. Finally, the program combined with message level table and put forward an efficient hierarchical group signature scheme based on elliptic curve. The analysis shows that the scheme greatly shortens the hierarchical group signature and verification time. Compared with existing schemes, the new scheme has high efficiency and greater safety. It also has strong practicability and the advantages of the elliptic curve cryptosystem. What's more, it is applicable to intelligent systems.
Key words: hierarchical group signature; group signature; elliptic curve; permissions; knowledge signature

1991 年,群签名概念由 Chaum 等人[1]提出。在一个群签名方案中,一个群体中的任意一个成员可以代表整个群体对消息进行签名,当发生争议时,群管理员能够确定签名者的真实身份。现有的大部分群签名方案均假设群成员具有相同的权限,但不能解决电子商务和电子政务中常出现的这种情况:公司管理层的成员经常需要履行工作职责内的权限,且只签署自己权限范围内的文件。李敏等人[2]首次提出了分级群签名方案,群成员具有不等的签名权限,任何群成员均不能对一个超出自己签名权限的消息产生有效的签名,验证者可以通过公开的消息权限标志来验证签名,方案利用双线性对构造了一个基于身份的群签名方案,签名验证时需要幂运算及多次模乘、点乘运算,方案效率有待提高。本文通过对椭圆曲线签名方案进行改进,避免了模逆、模乘运算及减少了点乘运算,缩短了签名时间和验证时间;在椭圆曲线数字签名的基础上,构造了一个基于椭圆曲线的高效的群签名方案,并引入一个消息等级表,将预定的消息等级和对应级别存入此表中;群成员的签名能力在加入群的时候由群管理员赋予,一个群成员在对一个消息进行签名时,以零知识的方法证明他所具有的签名权限,提出了一个基于椭圆曲线的高效分级群签名方案。该方案具有密钥短、运算速度快、存储空间小、占用带宽小等优点,适用于智能系统中。

1 准备工作

1.1 分级群签名定义及性质

群签名方案的定义及性质见文献[3]。类似分级群签名的定义及性质如下:

a)创建算法(SETUP)。能够产生群的公开密钥 Y 与群管理员的秘密密钥 S。

b)注册协议(JOIN)。群管理员与用户之间的一个协议,使得用户成为一个新的群成员。协议的输出为一个群成员的身份证书与一个相应的秘密密钥。

c)授权过程(AUTHORIZE)。权限管理员(群管理员)构造消息等级表,并根据成员的实际权限,给群成员分配权限 x_i,他可以签署相对应的消息等级 $p_i = g^{x_i}$。本文将利用知识签名来证明一个群成员具有的权限,任何成员都不能超越自己的权限签名,即使签了名,签名也不会被认可。

d)签名算法(SIGN)。当输入一个消息 m 与某个群成员的身份证书和秘密密钥后,输出对消息 m 的群签名。

e)验证算法(VERIFY)。当输入消息 m、消息的签名和群的公开密钥 Y 后,输出关于签名有效性的判断。

f)打开算法(OPEN)。当输入消息 m、消息的签名和群管

收稿日期: 2013-03-27; **修回日期**: 2013-05-15

作者简介:王国才(1963-),男,副教授,硕导,主要研究方向为计算机通信保密、计算机网络技术应用与网络信息安全、信息系统工程;刘美兰(1988-),女,硕士研究生,主要研究方向为网络信息安全(421858101@qq.com).

理员的秘密密钥 S 后,输出签名者的身份。

分级群签名应具有如下性质,其不可伪造性要比标准群签名强些,无关联性相对弱些:

a)不可伪造性(unforgeability)。在不知道群成员私钥或没有得到所需授权的条件下,任何攻击者都不可能成功伪造一个有效的授权群签名。

b)匿名性(anonymity)。除群管理员之外,任何人要确定一个给定群签名的实际签名人在计算上是不可行的。

c)可跟踪性(traceability)。必要时群管理员可以打开一个签名以确定签名人的身份,而且签名人不能阻止有效签名的打开。

d)无关联性(unlinkability)。在不打开签名的情况下,确定两个不同的群签名是否为同一个签名人所签是不可能的,但两个签名是否为具有同一权限的签名人所签是可知的。

e)防陷害性(exculpability)。包括群管理员在内的任何人都不能以其他群成员的名义产生有效的群签名。

f)抗联合攻击(coalition-resistance)。即使一些群成员串通在一起也不能产生一个有效的不能被跟踪的群签名。

如果某权限只授予一个成员,那么用该权限所签的多个签名就不是无关联的,但签名仍是匿名的;如果同一成员可授予多个不同的权限,那么不同权限的签名也不能排除是同一成员所签。

1.2　知识签名

知识签名是签名者在非交互的情况下向别人证明其知道某个秘密而不泄露该秘密本身。现在知识签名广泛应用在群签名中。本文采用的知识签名是基于 Schnorr 结构的[4,5]。

设 $G=\langle g\rangle$ 是阶为 n 的循环群,g 是 G 的生成元,y 是 G 中一个元素,以 g 为基的 y 的离散对数是使得 $g^x=y$ 成立的最小正整数 x。

定义 1　满足 $c=H(m\|y\|g\|g^s y^c)$ 的对 $(c,s)\in\{0,1\}^k\times Z_n^*$,称为元素 $y\in G$ 的以 g 为基的关于消息 m 的离散对数知识签名,表示为 SKREP[(α);$y=g^\alpha$](m)。如果秘密值 $x=\log_g(y)$ 已知,则这样的签名是容易计算的。从 Z_n^* 中随机地选择 r、c 和 s,可以这样计算:

$$c=H(m\|y\|g\|g^r),s=r-cx(\bmod n)$$

如果不知道秘密值 $x=\log_g(y)$,则计算这样的知识签名是困难的。本文将利用知识签名来证明群成员具有某种权限,同时并没有泄露该值。

1.3　消息等级表[2]

1)成员权限的划分　一个群可以设想为一个机构或组织,具有客观上的等级划分,可以根据这种客观的等级来构造一棵树。树的每一层代表一种权限,树的每一层可以有多个成员,也就是不同的成员可以有相同的权限,上下级关系映射为树的父子关系。

2)权限的分配　方案中有权限管理者,他可以是群管理员或者独立的第三方,主要负责消息等级表及权限的分配。假定群中共有 s 种权限,设循环群为 G,阶为 q,g 是一个随机选取的 G 的生成元。权限管理者(本方案为群管理员)随机从 z_q^* 选择不相等的数 $x_1,x_2,\cdots,x_s$,计算 $g^{x_i}=p_i(i=1,2,\cdots,s)$,将这 s 个数与权限一一对应起来。当一个成员注册时,群管理员根据成员的实际权限来分配他适当的 x_i 作为其权限。

3)消息等级的划分　消息等级的划分与权限的划分密切相关,即如果一个群成员具有权限标志 x_i,那么他可以签署的所有消息等级标志均为 $p_i=g^{x_i}$。消息区分表是一些数据记录,保存着消息特征及其对应等级的信息。签名的验证者可以访问这些数据记录获得消息的等级。其示意图如图 1 所示。

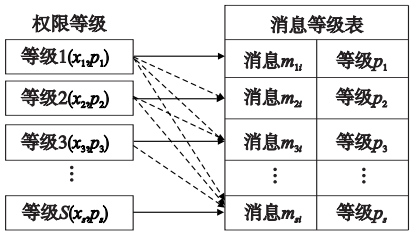

图1　消息等级表

图 1 中第一等级的权限为 x_1,第二等级的权限为 x_2,第三等级的权限为 x_3,本文群签名方案中,具有第一等级权限的成员 x_1 可以对消息 m_{1i} 产生有效签名。图 1 中实线表示具有较高等级权限的成员也不能对低等级的消息签名,因为权限分配设定了特定等级的消息必须由特定权限成员来签名;虚线表示高级的成员可以对较低级的消息进行签名,可根据实际情况来选择权限分配方法。本方案只对实线方案进行分析。

2　椭圆曲线数字签名方案及其改进

椭圆曲线密码中最著名的就是椭圆曲线数字签名算法(ECDSA)[6]。本文对 ECDSA 进行改进,避免其签名产生过程和签名验证过程中费时的求逆运算,从而一定程度上提高了签名生成和验证的速度。

2.1　椭圆曲线数字签名算法(ECDSA)

1)系统初始化阶段

a)选择一个在 FP 上的椭圆曲线 E,取属于 E 的一点 P,阶为 n。

b)取整数 $d\in[1,n-1]$。

c)计算 $Q=dP$。

2)签名产生阶段

a)取整数 k,使 $k\in[1,n-1]$。

b)计算 $kP=(x,y)$,$R=x \bmod n$。若 $R=0$,则转到 a)。

c)计算 $k^{-1} \bmod n$。

d)计算消息的散列值 $e=h(m)$,$S=k^{-1}(e+dR) \bmod n$。若 $S=0$,则转 a)。

3) 签名验证阶段

a)取得公钥(P,E,n,Q),验证 $R,S\in[1,n-1]$。

b)计算 $e=h(m)$。

c)计算 $t=S^{-1} \bmod n$。

d)计算 $u=et \bmod n$ 和 $u'=rt \bmod n$。

e)计算 $uP+u'Q=(x_1,y_1)$。

f)计算 $V=x_1 \bmod n$。

g)验证 $V=R$,若成立,签名有效;否则签名无效。

2.2　改进的椭圆曲线数字签名算法

1)系统初始化阶段

a)选一个在 FP 上的椭圆曲线 E,取属于 E 的一点 G,阶为 n。

b)取整数 $d\in[1,n-1]$。

c)计算 $Q=dG$。

2)签名产生阶段

a)取整数 k,使 $k\in[1,n-1]$。

b)计算 $kG=(x,y)$,$R=x \bmod n$。若 $R=0$,则转到 a)。

c)计算消息的散列值 $e=h(m\|R)$。

d)$S=(e+d-k) \bmod n$。若 $S=0$,则转 a)。

3)签名验证阶段

a)取得公钥(G,E,n,Q),验证$R,S\in[1,n-1]$。

b)计算$e=h(m\parallel R)$。

c)$X=(e-S)G+Q=(x_1,y_1)$。

d)计算$V=x_1 \bmod n$。

e)验证$V=R$,若成立,签名有效;否则签名无效。

2.3 改进算法正确性证明

若消息M的签名(R,S)是由合法的签名者生成的,先计算$e=h(m\parallel R)$,$S=(e+d-k) \bmod n$,则$k=(e+d-S) \bmod n$。于是:

$$X=u_1G+Q=(e-S)G+Q=eG-SG+dG=(e+d-S)G=kG$$

则$V=R$,即签名是合法的。

2.4 改进前后方案效率对比

算法效率主要考虑模逆、模乘、点乘运算,其他可以忽略。改进前后方案效率对比如表1所示。

表1 改进前后方案效率对比

方案	签名阶段运算量	验证阶段运算量	总计
原 ECC	1 次模逆运算 1 次模乘运算 1 次点乘运算	1 次模逆运算 2 次模乘运算 2 次点乘运算	2 次模逆运算 3 次模乘运算 3 次点乘运算
改进后	1 次点乘运算	1 次点乘运算	2 次点乘运算

从算法运算量角度分析,如果模乘运算的数据规模为n,1次点乘运算复杂度为$O(n^2\log_2 n)$,1次模乘运算复杂度为$O(n^2)$,1次模逆的运算量相当于9次模乘[7],则由表1可知,原椭圆曲线数字签名方案总运算量为2次模逆运算、3次模乘运算、3次点乘运算,则总时间复杂度为$(3\log_2 n+21)n^2$。改进后椭圆曲线数字签名方案总运算量为2次点乘运算,则总时间复杂度为$2n^2\log_2 n$。根据表1分析可得,本方案避免了耗时的模逆和模乘运算,并减少了一次点乘运算,方案运算速度明显提高了,缩短了签名和验证时间,有效提高了算法的效率,所以此椭圆曲线方案是一个高效的方案。

3 基于椭圆曲线的分级群签名方案

3.1 系统初始化

系统初始化由群管理员完成,取一个大素数p,随机取$a,b\in Z_p$,使$4a^3+27b^2\neq 0$,构造一条椭圆曲线$y^2=x^3+ax+b$,满足点数$\#E(Z_p)$能被一个大素数$n(n\geqslant 2^{160})$整除,取$E(Z_p)$上一个阶为n的点G作为基点,H是一个安全的hash函数,Ψ表示将椭圆曲线上的点$P(x,y)$转换为x,记为$(P)_x$,群管理员A取一个私钥$k_A\in Z_n^*$,计算公钥$K_A=k_AG$,秘密保存私钥k_A,其他参数p、a、b、n、G、Y_A和H均公开。

3.2 成员注册及授权

a)新成员B随机取k_B,计算$K_B=k_BG$,其中私钥为k_B,公钥为K_B,然后将K_B和新成员B的身份标志ID_B发给群管理员A。

b)群管理员A随机选择$u\in Z_n^*$,计算$\mathrm{ID}_c=H(u\parallel \mathrm{ID}_B)G$,群管理员将$\mathrm{ID}_C$秘密发给成员$B$,$\mathrm{ID}_C$绑定了成员$B$的身份,成员$B$用$\mathrm{ID}_C$产生和发布群签名。

c)群管理员保存每一个新成员的三元素,即保存$(\mathrm{ID}_B,\mathrm{ID}_C,u)$,当发生争议时,可以确定签名者的真实身份。

d)群管理员A随机取$v\in Z_n^*$,计算

$$V=vG\neq 0,s_A=k_AH((\mathrm{ID}_C)_x\parallel (V)_x)+v(\bmod n)$$

e)群管理员A将(V,s_A)发给B,同时群管理员给新成员B一个适当的用户权限(x_i,p_i),其中$p_i=g^{x_i}$,新成员B的身份证书为$(K_B,\mathrm{ID}_C,V,s_A,(x_i,p_i))$。

3.3 签名产生

a)对于权限为p_i的消息m,群成员B根据自己的成员证书和权限执行以下操作:

随机选择$r\in Z_n^*$,计算

$R=rG\neq 0$

$e=H(m\parallel (R)_x)$

$s=e+k_B-r \bmod n$

$I=\mathrm{ID}_C+\mathrm{ID}_B+k_BK_A$

$SK=\mathrm{SKREP}[(\alpha);p_i=g^{\alpha}](m\parallel R\parallel e\parallel s\parallel I)$

b)群成员B将群签名$\sigma=(m,s,R,I,K_B,SK)$发送给签名验证者。

3.4 签名验证

a)验证者首先查询等级表获得消息m的等级标志p_i。

b)验证者验证等式:

$$s_AG=H((\mathrm{ID}_C)_x\parallel (V)_x)K_A+V$$

若等式成立,则证明签名是群中成员的签名,否则拒绝签名。

c)验证者验证$K_B=(s-e)G+R$,如果等式成立,且$c=H(m\parallel p_i\parallel g\parallel g^sp_i^c)$,则接收签名$\sigma=(m,s,R,I,K_B,SK)$,否则拒绝签名。

3.5 签名打开

当发生争议时,可以撤销群成员的匿名性,打开签名,确定签名者的真实身份。

a)群管理员A计算

$$E=I-k_AK_B,F=I-ID_C-k_AK_B$$

b)群管理员A查询保存的三元素,找到$F=\mathrm{ID}_B$,$E=\mathrm{ID}_B+\mathrm{ID}_C$,则$\mathrm{ID}_B$就为真实签名者的身份。

c)群管理员发送(ID_B,u)给签名验证者,计算$\mathrm{ID}_C'=H(u\parallel \mathrm{ID}_B)G$,若$\mathrm{ID}_C=\mathrm{ID}_C'$,则证明成功确定了群签名者的真实身份。

4 方案的安全性分析

1)匿名性和不可链接性 攻击者要确定ID_B的身份信息,必须通过计算$\mathrm{ID}_c=H(u\parallel \mathrm{ID}_B)G$,面临着解椭圆曲线离散对数难题(ECDLP)和单向哈希函数问题。u是群管理员随机选取的参数,攻击者不能确定绑定身份ID_C和真实身份ID_B,群成员签名是用绑定身份ID_C,攻击者不能确定真正签名者的身份,因此,群签名方案具有匿名性和不可链接性。

2)不可伪造性 假设攻击者(群外的攻击者、群管理员和群中其他成员)想伪造签名,必须同时知道随机数r、签名密钥k_B和用户权限x_i,其中随机数r是随机选取的,攻击难度较大;签名密钥k_B必须通过计算等式$K_B=k_BG$才能获得,这将会面临椭圆曲线离散对数难题(ECDLP),在计算上是行不通的;获取用户权限x_i时,必须计算$p_i=g^{x_i}$,这将会面临离散对数难题,在计算上也是不可行的。综上所述,分级群签名方案具有不可伪造性。

3)抗联合攻击 如果群管理员和群中其他有效成员想联合伪造签名,他们必须知道签名私钥k_B和随机参数r,在计算上会面临椭圆曲线离散对数难题,在计算上是不可行的,所以方案具有抗联合攻击性。

4)可跟踪性和不可否认性 当发生争议时,群管理员可以通过计算$E=I-k_AK_B$和$F=I-\mathrm{ID}_C-k_AK_B$来确定群成员的

真实身份,因此匿名性是可以追踪的,签名者也不可否认自己的签名。

5)抗越权性　每一个群成员想对超出自己权限的消息产生有效的签名,显然他面临的是产生正确的知识签名 SK,面对解离散对数问题难题,所以本方案不能对超出自己权限的消息产生有效的签名。

5　方案的性能分析与比较

下面将本文提出的方案与现有的分级群签名方案进行比较,比较过程中只考虑计算量较大的点乘、标乘、双线性对运算,其他运算对整个方案运算量的比较没有太大的影响,因此可忽略不计。文献[2]中的分级群签名方案签名过程中的运算量为 1 次点乘运算、10 次乘法运算,签名验证过程中的运算量为 3 次双线性对运算,签名打开过程中的运算量为 3 次双线性对运算,因此总运算量为 1 次点乘运算、10 次乘法运算、6 次双线性对运算。本文提出的新方案签名过程中的运算量为 1 次点乘运算、1 次乘法运算,签名验证过程中的运算量为 2 次点乘运算、1 次乘法运算,签名打开过程中的运算量为 2 次标乘,因此新方案总运算量为 3 次点乘运算、4 次乘法运算。由文献[8]中的表 4.3 可知,当嵌入次数 $k=6$,大素数 $|r|=160$ 时,Tate 双线性对算法效率为 54 758 次点乘运算。则文献[2]中分级群签名方案的运算量远远超过本文所提出的新方案,由此可知新方案节省了大量运算时间,具有高效性。此外加入权限等级表和知识签名证明权限,对方案效率不产生太大的影响,并且方案是基于椭圆曲线的,因此方案还具有密钥短、高效、占用存储空间小、对宽带需求小、系统开销小等优点。

6　结束语

本文通过对椭圆曲线群签名方案进行改进,签名算法和验证算法避免了模逆和模乘运算,并减少了 1 次点乘运算,提高了方案签名和验证算法效率,并通过引入权限等级表,提出了一个基于椭圆曲线的高效分级群签名方案。与现有的方案对比,其大大节约了运算时间,计算量小,处理速度快,且具有安全性能高、存储空间占用小、对带宽要求低等优点,方案解决了特定的应用环境,实用性强,在智能系统中有着广泛的应用前景。

参考文献:

[1] CHAUM D, Van HEYST E. Group signatures[C]//Proc of EUROCRYPT'91. Berlin: Springer-Verlag, 1991: 257-265.

[2] 李敏, 王尚平, 马晓静, 等. 分级群签名[J]. 计算机应用研究, 2006, 23(9): 88-91.

[3] 薛春生. 群签名方案的研究[D]. 南京: 南京航空航天大学, 2007.

[4] PARK S, KIM S, WON D. ID-based group signature[J]. Electronics Letters, 1997, 33(19): 1616-1617.

[5] 汤鹏志, 李彪. Schnorr 数字签名的零知识证明[J]. 微电子学与计算机, 2012, 29(6): 177-179.

[6] JOHNSON D, MENEZES A, VANSTONE S. The elliptic curve digital signature algorithm (ECDSA) [J]. International Journal of Information Security, 2001, 1(1): 36-63.

[7] 韩笑, 施荣华. 一种高效的椭圆曲线数字签名方案[J]. 微计算机信息, 2012, 28(9): 395-397.

[8] 魏鹏娟. 椭圆曲线的选取与双线性对的快速计算研究[D]. 西安: 西安电子科技大学, 2011.

(上接第 585 页)编结果中对数据拷贝类函数调用来判断是否可能发生溢出。然而程序的反汇编结果缺少变量信息,并且程序结构不明确,使用简单的模式匹配会导致大量误报。Hex-Rays 反编译结果恢复了部分变量信息,程序结构清晰,在此基础上进行漏洞挖掘工作有很大优势。本文提出的方法对反编译结果进行处理,在模式匹配检测结果的基础上,借助数据之间的关联关系对初步检测结果进行筛选,在一定程度上减少了使用简单模式匹配所造成的误报。

从表 1 中可以看出,使用本文提出的方法后,检测结果的精度得到了进一步的提高。对于漏洞挖掘人员来说,缩小了代码分析范围,使得分析目标更加明确,提高了漏洞挖掘的效率。

4　结束语

本文将数据关联性分析应用到漏洞挖掘中,并开发出了相应的插件。该插件以目标二进制文件为输入,借助 Hex-Rays 反编译插件生成每个函数对应的类 C 代码,然后构造函数的 AST,在 AST 之上作漏洞挖掘。使用该插件可以大大提高漏洞挖掘人员对缓冲区溢出类漏洞的挖掘效率。本方法虽然在一定程度上提高了漏洞挖掘结果的精度,但还存在明显的不足。当前,本文所做的检测主要是针对过程内,并不关心过程间数据传递及控制流转移,下一步需要对这些方面作进一步研究。此外,插件检测结果的精确性需要进一步提高,还需对检测方法进行优化。

参考文献:

[1] CNCERT. 2011 年我国互联网网络安全态势综述[EB/OL]. (2012-03-19)[2012-08-01]. http://www.cert.org.cn/UserFiles/File/201203192011annualreport(1).pdf.

[2] CERT Coordination Center. CERT/CC statistics[EB/OL]. (2011). http://www.cert.org/stats/.

[3] KUANG Chun-guang, MIAO Qing, CHEN Hua. Analysis of software vulnerability[C]//Proc of the 5th WSEAS International Conference on Information Security and Privacy. [S.l.]: World Scientific and Engineering Academy and Society, 2006: 218-223.

[4] REDWINE S, DAVIS N. Processes to produce secure software[EB/OL]. (2004-03). http://www.cigital.com/papers/download/secure-software-process.pdf.

[5] BALAKRISHNAN G. WYSINWYX: what you see is not what you execute[D]. Wisconsin: University of Wisconsin, 2007.

[6] 张晓锋. 软件逆向工程相关技术研究与实现[D]. 成都: 电子科技大学, 2007.

[7] TAKANEN A, De MOTT J, MILLER C. Fuzzing for software security testing and quality assurance [M]. London: Artech House, 2008.

[8] YUAN Jing-bo, DING Shun-li. A method for detecting buffer overflow vulnerabilities[C]//Proc of the 3rd IEEE International Conference on Communication Software and Networks. 2011: 188-192.

[9] RAWAT S, MOUNIER L. Finding buffer overflow inducing loops in binary executables[C]//Proc of IEEE International Conference on Software Security and Reliability. 2012: 177-186.

[10] 忽朝俭, 李舟军, 郭涛, 等. 写污点值到污点地址漏洞模式检测[J]. 计算机研究与发展, 2011, 48(8): 1455-1463.

[11] BELLETTINI C, RRUSHI J L. Vulnerability analysis of SCADA protocol binaries through detection of memory access taintedness[C]//Proc of IEEE Workshop on Information Assurance. 2007.